REMARKS

Claims 1, 8, and 13 stand objected to because of noted informalities. Claims 13-18 stand rejected under 35 USC §112, first paragraph. Claims 1-4, and 7-18 stand rejected under 35 USC §103(b) as being unpatentable over Lee, U.S. patent 7,088,689 in view of Kirchner et al., U.S. patent 6,263,370 and Smyk, U.S. patent 6,289,001.

Claims 1, 8, and 13-18 have been amended to more clearly state the invention and to accommodate the Examiner's objections.

Reconsideration and withdrawal of the objections to claims 1, 8, and 13, and the rejection under 35 USC §112, is respectfully requested.

Reconsideration and allowance of the pending claims 1-4, and 7-18, as amended, is respectfully requested.

Lee, U.S. patent 7,088,689 discloses a virtual local area network(VLAN) data switching method uses an address resolution protocol(ARP) packet in VLAN constructed with one internet protocol(IP) subnet. This is achieved by making a source host broadcast an ARP request packet, making a Proxy ARP server prepare to transmit a virtual ARP request packet wherein the Proxy ARP server receives the ARP request packet, and making a destination host transmit an ARP response packet in direct to the source host through a switching unit wherein the destination host receives the virtual ARP request packet. Data transfer delay time for the data exchanged between different VLANs as well as the router load occurring in handling the traffic between VLANs are reduced, so that data handling performance for high-capacity multimedia data

transferred on VLANs is improved. Column 2, lines 19-25 states:

An object of the present invention is to provide a VLAN data switching method using ARP packets which enable routing procedures to be minimized, which occurs when transferring data between different VLANs in a VLAN environment constructed with one IP subnet, by employing virtual ARP requests to achieve a direct switching operation by a switch.

Column 2, lines 45-54 states:

In another aspect of the present invention, a VLAN data switching method using an ARP packet includes the steps of making a source host broadcast an ARP request packet, making a Proxy ARP server prepare to transmit a virtual ARP request packet wherein the Proxy ARP server receives the ARP request packet, and making a destination host transmit an ARP response packet directly to the source host through a switching unit wherein the destination host receives the virtual ARP request packet.

Column 3, lines 34-41 states:

Moreover, the ARP recognizes a MAC address of a destination host or of a gateway using a broadcasting for the IP address of the destination. A collected IP address and the corresponding MAC address are stored in a memory(i.e., an ARP cache) of each IP host in a table form so as to be used for a next packet transmission procedure. Therefore, the ARP always checks the ARP cache for the IP address and MAC address prior to requesting the MAC address.

Column 3, line 66 through column 4, line 12 states:

In this case, the Proxy ARP server 10a stores a source A of an inter-LAN data, and an IP address and a MAC address of a destination host B, generates a virtual ARP request packet including IP and MAC addresses of the destination host A, and transmits the virtual ARP request packet to the destination host B, whereby the destination host B receiving the corresponding packet enables the ARP request packet to be transferred directly to the source host A. Namely, the ARP request packet is transferred by a switching of two adjacent layers through the Proxy ARP server 10a and the switching unit 20. In this case, the Proxy ARP server 10a includes a MAC table(not shown in the drawings) storing the MAC and IP addresses for the hosts belonging to the respective VLANs.

Kirchner et al., U.S. patent 6,263,370 discloses a TCP/IP-based client-server interface for sending service requests from a client computer to a Network Information Distribution Services server. The TCP/IP-based client-server interface is an

easily implemented and economical alternative to the proprietary UDP/IP-based client-server interface developed for communication between client computers and Network Information Distribution Services servers. A service request is sent to a specific TCP/IP logical port associated with a particular type of service. A process associated with the specific TCP/IP logical port formats the request in the same manner as requests received through the proprietary UDP/IP-based client-server interface are formatted, and then directs the formatted requests to the Network Information Distribution Services server process that executes the requests. Column 10, lines 22-43 states:

> FIG. 8 displays the flow of messages through the various internal interfaces of the TCP/IP-based client-server interface. Comparison of FIGS. 7 and 8 reveals the markedly different sequencing and flow of messages between clients and NIDS servers using the NSPP-based client-server interface and clients and servers using the TCP/IP-based client-server interface. A first difference is that, in order to make a connection to a specific NIDS service and to request execution of an operation by that service, a client using the TCP/IP-based client-server interface needs to send only two requests. The first request is a standard TCP/IP connection request 801. The client application calls the NIDS API 802 to make the connection. The NIDS API calls a well-known TCP/IP function 803 to establish a connection with the logical port on the NIDS server that corresponds to the service requested through the NIDS API. The client's TCP/IP socket transmits the TCP/IP connection request to the server's TCP/IP logical port corresponding to the NIDS service to which the client wishes to connect 804. The TCP/IP socket corresponding to that logical port passes the message to the NIDSCOM process associated with the service 805. The NIDSCOM process then conducts a dialog with the NIDS process on the NIDS server that handles the UDP/IP-based NSPP interface. This is the same NIDS process that executes the first two requests in FIG. 7. In fact, the NIDSCOM serves as a proxy or agent on the NIDS server for the client. The NIDSCOM process executes the same open_session and pick_request requests 806-809 made by a client under the NSPP-based client-server interface shown in FIG. 7

Smyk, U.S. patent 6,289,001 discloses asynchronous transfer mode ("ATM") proxy signaling methods and apparatus that provide ATM signaling reliability. A plurality of proxy agents are connected to a ATM switch. The ATM switch proxy agents are connected via SVCs established under the control of a proxy agent selector. The

proxy agent selector identifies an alternative proxy agents should one or more of the other proxy agents fail. Upon failure of one or more proxy agents, under the control of timing operations within the controller of the switch, the proxy agent selector selects one or more alternative proxy agents. In this manner, proxy agent signaling continues to a ATM switch undisturbed.

The present invention overcomes a problem of a Virtual IP Address (VIPA) on a server TCP/IP introduced by International Business Machines Corporation that has been used on some server computers, such as an iSeries server manufactured and sold by International Business Machines Corporation. A VIPA is configured generally the same as a normal IP address for a physical adapter, except that it is not associated with any particular device. To an attached router, the TCP stack on the server computer looks like another router. When the TCP stack receives a packet destined for one of its VIPAs, the inbound IP function of the TCP stack notes that the IP address of the packet is in the TCP stack's home list of IP addresses and forwards the packet up the TCP stack. The home list of a TCP stack is the list of IP addresses that are owned by the TCP stack. Assuming the TCP stack has multiple adapters or paths to it, if a particular physical adapter fails, the attached routing network will route VIPA-targeted packets to the TCP stack via an alternate route. The VIPA may, thus, be thought of as an address to the stack, and not to any particular adapter. However, a problem remains because the Virtual IP addresses are not directly routable. In other words, the iSeries server will never answer an ARP request destined to Virtual IP address. That is why the local gateways or routers needed explicit routes configured in order to forward packets

to the Virtual IP interface. But, the downside is that configuring these same additional explicit routes configured in order to forward packets to the Virtual IP interface in each locally attached client is cumbersome at best, and impractical at worst. The local clients could be configured to send all of their data to the routers, but that just adds unnecessary traffic to the router. Or the local clients could access the iSeries server using one of the physical IP addresses, but if the adapter on which that IP address is defined goes down, the iSeries server appears down to the client. The net result that, for locally attached clients, the fault tolerance, high availability advantages of Virtual IP are not available.

As defined in the IBM Dictionary of Computing, a subnet in TCP/IP is a part of a network that is identified by a portion of the Internet address. The address to all nodes in a subnet starts with the same binary sequence.

As recited in independent claims 1, 8, and 13, as amended, the present invention provides a method, apparatus and computer program product for implementing enhanced proxy Address Resolution Protocol (ARP) for Virtual Internet protocol (IP) addresses. The claimed invention as recited in independent claims 1, 8, and 13, as amended, is not shown or disclosed in the Lee patent, nor rendered obvious by the combined teachings of all.

Independent claim1 recites the steps of identifying a Virtual Internet protocol (IP) interface requiring proxy ARP, dynamically selecting a proxy agent for said Virtual Internet protocol (IP) interface; said selected proxy agent and said Virtual Internet protocol (IP) interface being in a same subnet; adding an IP address for said

Virtual Internet protocol (IP) interface to an address list of an associated physical adapter for said selected proxy agent; and utilizing said physical adapter for said selected proxy agent, broadcasting said added IP address for said Virtual Internet protocol (IP) interface with a media access control (MAC) address of said associated physical adapter for said selected proxy agent, responsive to failure of said selected proxy agent; dynamically selecting a new proxy agent for said Virtual Internet protocol (IP) interface by Transmission Control Protocol/Internet Protocol (TCP/IP) code; and wherein dynamically selecting said proxy agent for said Virtual Internet protocol (IP) interface includes checking for a proxy agent in the same subnet as said Virtual Internet protocol (IP) interface.

As amended, independent claim 1 further recites the subject matter of original dependent claims 5 and 6.

The subject matter of the invention and recited steps of amended independent claim 1 are not shown or disclosed, nor suggested in the references of record including Lee, Kirchner, and Smyk. The Lee patent discloses a VLAN data switching method using ARP packet. Lee does not show nor suggest identifying a Virtual Internet protocol (IP) interface requiring proxy ARP, as taught and claimed by Applicants. This directly routable configuration allows the iSeries server 100 to answer ARP requests, that is, to Proxy ARP, for Virtual IP addresses. Proxy ARP for Virtual IP interfaces provides customers with an increased fault tolerance, and higher system availability. A new "Proxy ARP Yes/No" option is presented when a Virtual IP interface is configured. This option defaults off to prevent existing configurations from breaking

upon upgrading the new support feature in accordance with the preferred embodiment. When this new enhanced Proxy ARP option is enabled, the Transmission Control Protocol/Internet Protocol (TCP/IP) code answers ARP requests to the Virtual IP address (VIPA).

The Examiner acknowledges that Lee does not disclose responsive to failure of the selected proxy agent, dynamically selecting a new proxy agent for the Virtual Internet protocol interface by TCP/IP code. Applicants teach this new support feature that is added to a server computer to allow Virtual IP addresses to be configured as directly routable. The Examiner acknowledges that Lee does not disclose providing TCP/IP code for dynamically selecting the proxy agent.

Lee does not teach or suggest dynamically selecting a proxy agent for said Virtual Internet protocol (IP) interface; said selected proxy agent and said Virtual Internet protocol (IP) interface being in a same subnet; adding an IP address for said Virtual Internet protocol (IP) interface to an address list of an associated physical adapter for said selected proxy agent; and utilizing said physical adapter for said selected proxy agent, and broadcasting said added IP address for said Virtual Internet protocol (IP) interface with a media access control (MAC) address of said associated physical adapter for said selected proxy agent, as taught by Applicants and recited in independent claim 1, as amended. Applicants respectfully submit that the additional cited references of record including Kirchner and Smyk add nothing to suggest the method of the invention, as recited in independent claim 1, as amended.

Thus, independent claim 1, as amended, is patentable.

Independent claim 8, as amended, recites apparatus for implementing enhanced proxy Address Resolution Protocol (ARP) for Virtual Internet protocol (IP) addresses comprising: a local network; a server computer having a Virtual Internet protocol (IP) address, a Virtual Internet protocol (IP) interface, and a plurality of physical adapters connecting said server computer to said local network, a Transmission Control Protocol/Internet Protocol (TCP/IP) code for dynamically selecting a proxy agent for said Virtual Internet protocol (IP) interface; said Transmission Control Protocol/Internet Protocol (TCP/IP) code being responsive to failure of said selected proxy agent; for dynamically selecting a new proxy agent for said Virtual Internet protocol (IP) interface by Transmission Control Protocol/Internet Protocol (TCP/IP) code including checking for a proxy agent in the same subnet as said Virtual Internet protocol (IP) interface; said selected proxy agent and said Virtual Internet protocol (IP) interface being in a same subnet; a proxy ARP for Virtual IP interface initiation task for adding an IP address for said Virtual Internet protocol (IP) interface to an address list of an associated one of said physical adapters for said selected proxy agent; and for utilizing said physical adapter for said selected proxy agent for broadcasting said added IP address for said Virtual Internet protocol (IP) interface with a media access control (MAC) address of said associated physical adapter for said selected proxy agent.

The present invention as taught and claimed in independent claim 8, as amended, the proxy agent is dynamically selected by the TCP/IP stack code. This feature is not disclosed, nor suggested by the references of record. An advantage of dynamically selecting the proxy agent by the TCP/IP stack code is so the proxy agent

interface is dynamic. If the interface that is currently acting as the proxy agent for a Virtual IP interface goes down, the agent function is immediately moved to an alternate interface, if one exists, so that the Virtual IP address remains accessible to local and remote clients. Thus, with this new support feature of dynamically selecting the proxy agent by the TCP/IP stack code allows Virtual IP addresses to be configured as directly routable. This directly routable configuration allows the iSeries server to answer ARP requests, that is, to Proxy ARP, for Virtual IP addresses. Proxy ARP for Virtual IP interfaces provides customers with an increased fault tolerance, and higher system availability.

Lee, Kirchner, and Smyk do not teach or suggest, the proxy ARP or Virtual IP interface initiation task for adding an IP address for said Virtual Internet protocol (IP) interface to an address list of an associated one of said physical adapters for said selected proxy agent; and for utilizing said physical adapter for said selected proxy agent for broadcasting said added IP address for said Virtual Internet protocol (IP) interface with a media access control (MAC) address of said associated physical adapter for said selected proxy agent, as further recited in independent claim 8, as amended.

Applicants respectfully submit that the cited references of record including Lee, Kirchner, and Smyk fail to suggest the apparatus of the invention, as recited in independent claim 8, as amended.

Thus, independent claim 8, as amended, is patentable.

Independent claim 13, as amended, recites a computer program product

for implementing enhanced proxy Address Resolution Protocol (ARP) for Virtual Internet protocol (IP) addresses in a server computer, said computer program product including instructions stored on a computer recording medium, wherein said instructions, when executed by the server computer to cause the server computer to perform the steps of: identifying a Virtual Internet protocol (IP) interface requiring proxy ARP; dynamically selecting a proxy agent for said Virtual Internet protocol (IP) interface;
adding an IP address for said Virtual Internet protocol (IP) interface to an address list of an associated physical adapter for said selected proxy agent; utilizing said physical adapter for said selected proxy agent, and broadcasting said added IP address for said Virtual Internet protocol (IP) interface with a media access control (MAC) address of said associated physical adapter for said selected proxy agent, responsive to failure of said selected proxy agent; dynamically selecting a new proxy agent for said Virtual Internet protocol (IP) interface by Transmission Control Protocol/Internet Protocol (TCP/IP) code; and wherein dynamically selecting said proxy agent for said Virtual Internet protocol (IP) interface includes checking for a proxy agent in the same subnet as said Virtual Internet protocol (IP) interface..

Independent claim 13, as amended, further recites the subject matter of original dependent claims 5 and 6, and is patentable for the same reasons as in independent claim 1. Only applicants teach the recited steps of independent claim 13, as amended.

Thus, each of the independent claims 1, 8, and 13, as amended, is patentable.

Dependent claims 2-4, 9-12, and 14-18 depend from respective patentable independent claims 1, 8, and 13, further defining the invention, and are likewise patentable.

Applicants have reviewed all the art of record, and respectfully submit that the claimed invention is patentable over all the art of record, including the references not relied upon by the Examiner for the rejection of the pending claims.

It is believed that the present application is now in condition for allowance and allowance of each of the pending claims 1-4, and 7-18, as amended, is respectfully requested. Prompt and favorable reconsideration is respectfully requested.

If the Examiner upon considering this amendment should find that a telephone interview would be helpful in expediting allowance of the present application, the Examiner is respectfully urged to call the applicants' attorney at the number listed below.

Respectfully submitted,

S-signature by

___/Joan Pennington/_____
By: Joan Pennington
Reg. No. 30,885
Telephone: (312) 670-0736

November 23, 2007